

Individual Performance Test #1

Documentation

By:

Ola, John Lloyd Cyrus

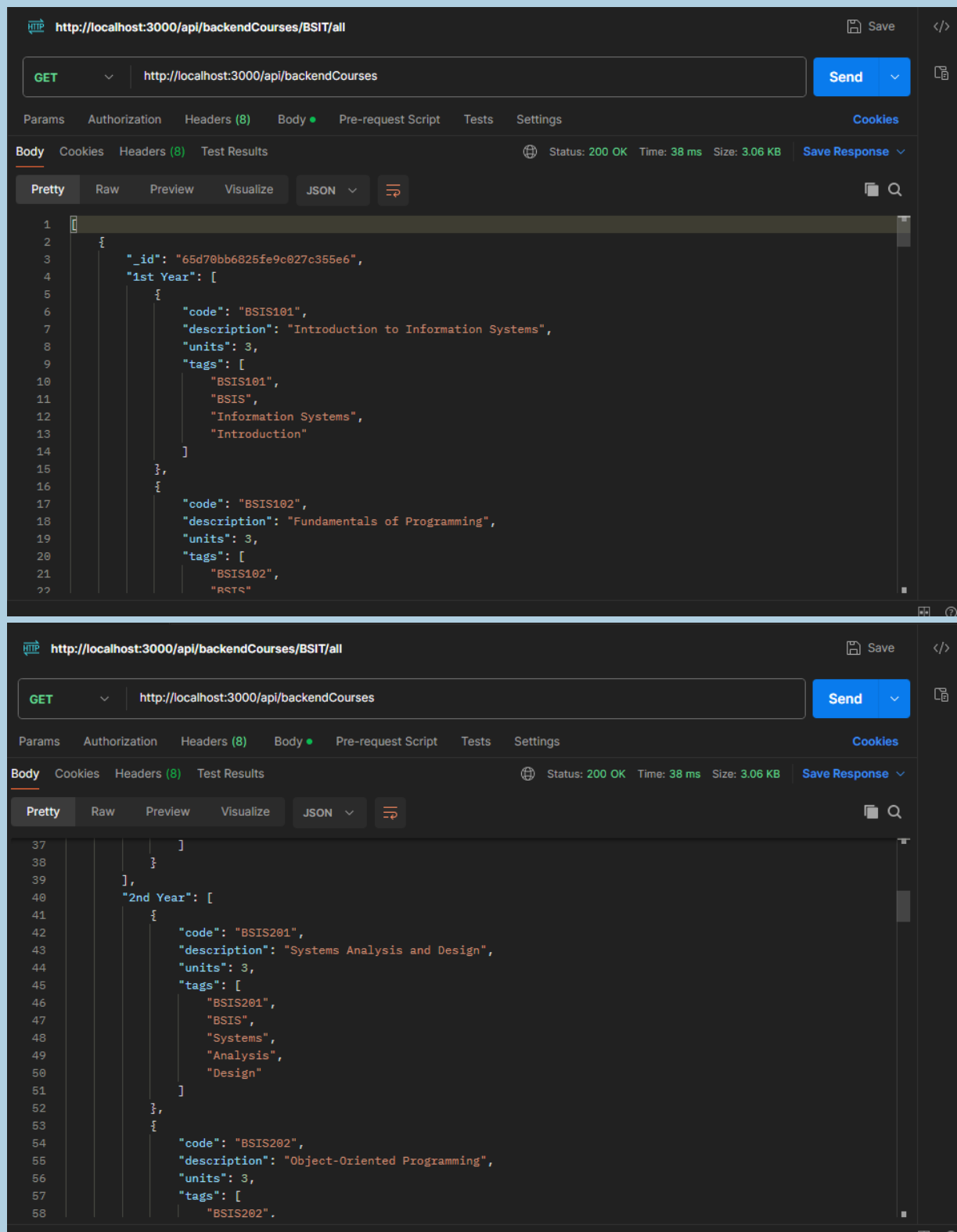
Teacher:

Villi Dane Go

23/02/2024

The Hardest thing about this is not the code or the logic but the MongoDB, at first I can't connect it and can't get result from it I started to try it on Feb 20, 2024 but can't make it work. But still to make it and make it run the way I intended.

Here It Retrieve all published backend courses from MongoDB



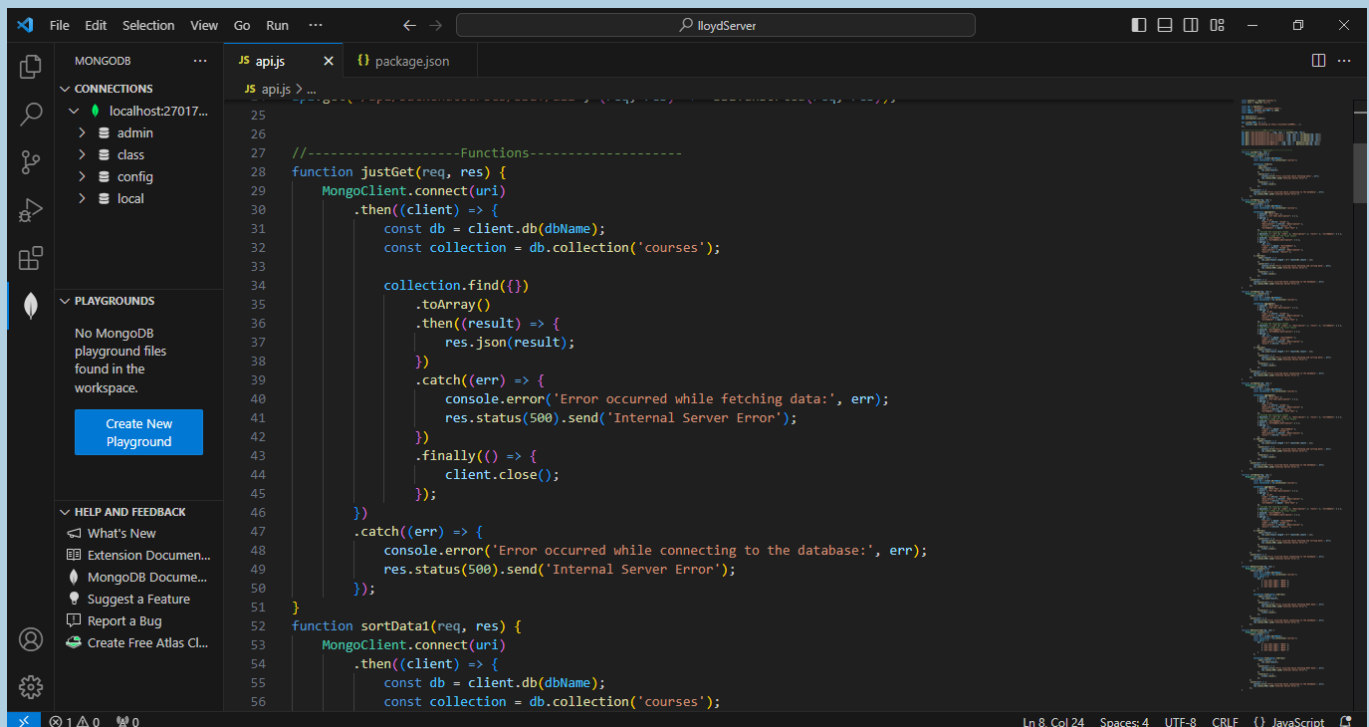
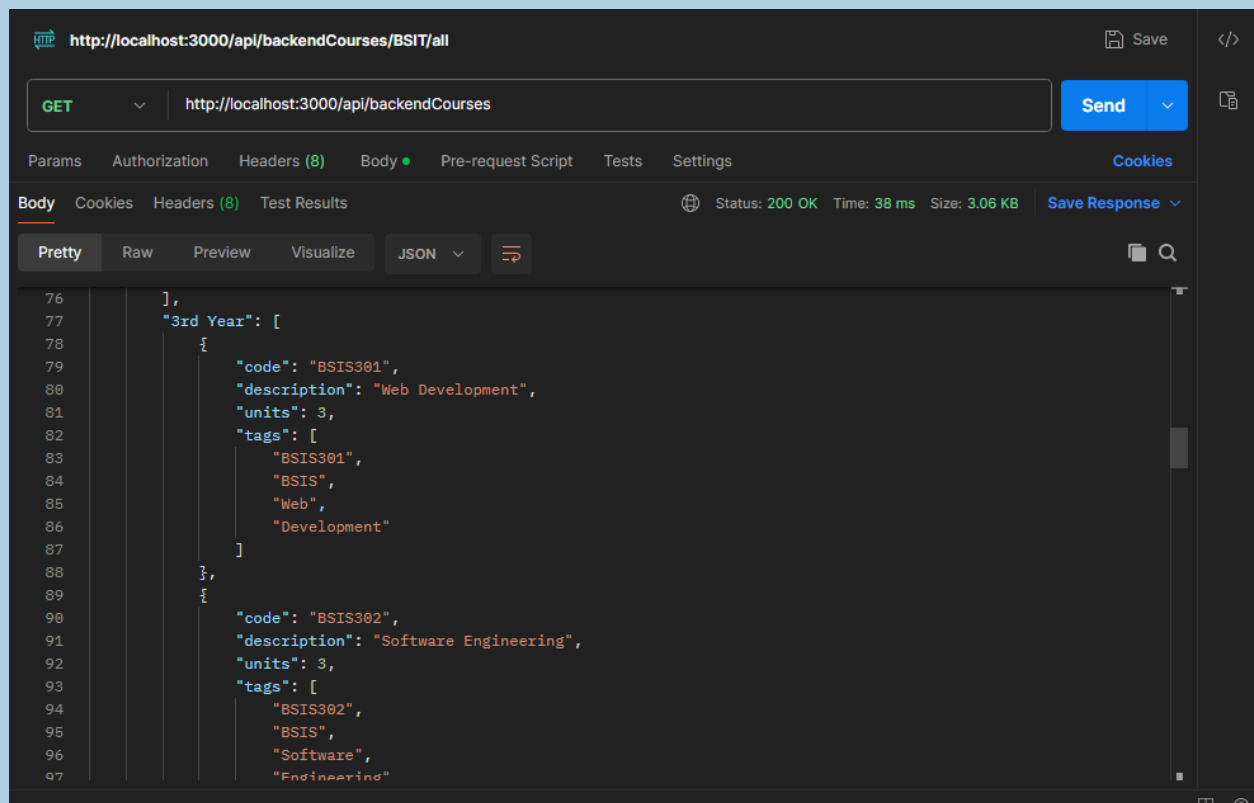
The image displays two screenshots of a REST client interface, likely Postman, showing the results of a GET request to the endpoint `http://localhost:3000/api/backendCourses/BSIT/all`. The status is 200 OK, and the response is in JSON format.

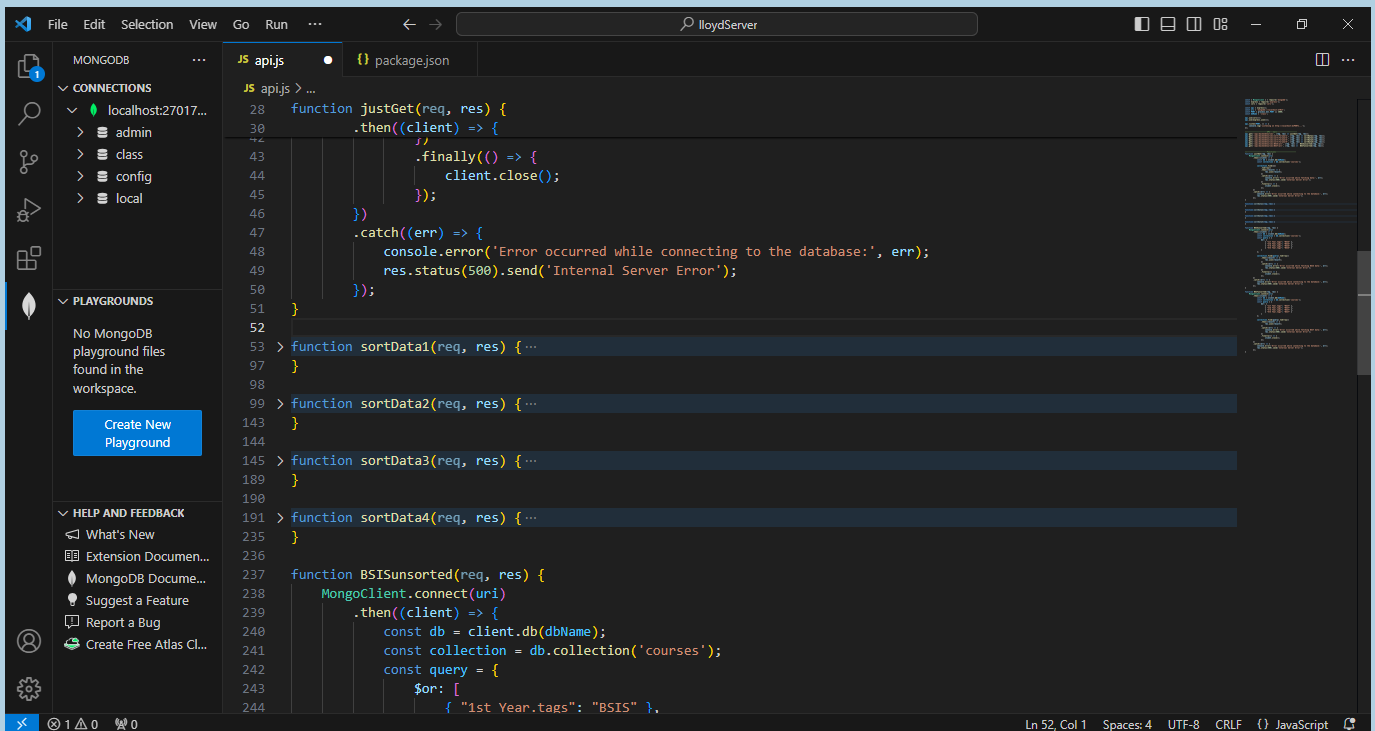
Top Screenshot: Shows the first part of the JSON response, including the first year of courses.

```
1 {
2   "_id": "65d70bb6825fe9c027c355e6",
3   "1st Year": [
4     {
5       "code": "BSIS101",
6       "description": "Introduction to Information Systems",
7       "units": 3,
8       "tags": [
9         "BSIS101",
10        "BSIS",
11        "Information Systems",
12        "Introduction"
13      ]
14    },
15    {
16      "code": "BSIS102",
17      "description": "Fundamentals of Programming",
18      "units": 3,
19      "tags": [
20        "BSIS102",
21        "BSIS"
22      ]
23    }
24  ]
25 }
```

Bottom Screenshot: Shows the continuation of the JSON response, including the second year of courses.

```
37 ]
38 },
39 "2nd Year": [
40   {
41     "code": "BSIS201",
42     "description": "Systems Analysis and Design",
43     "units": 3,
44     "tags": [
45       "BSIS201",
46       "BSIS",
47       "Systems",
48       "Analysis",
49       "Design"
50     ]
51   },
52   {
53     "code": "BSIS202",
54     "description": "Object-Oriented Programming",
55     "units": 3,
56     "tags": [
57       "BSIS202",
58       "BSIS"
59     ]
60   }
61 ]
62 }
```

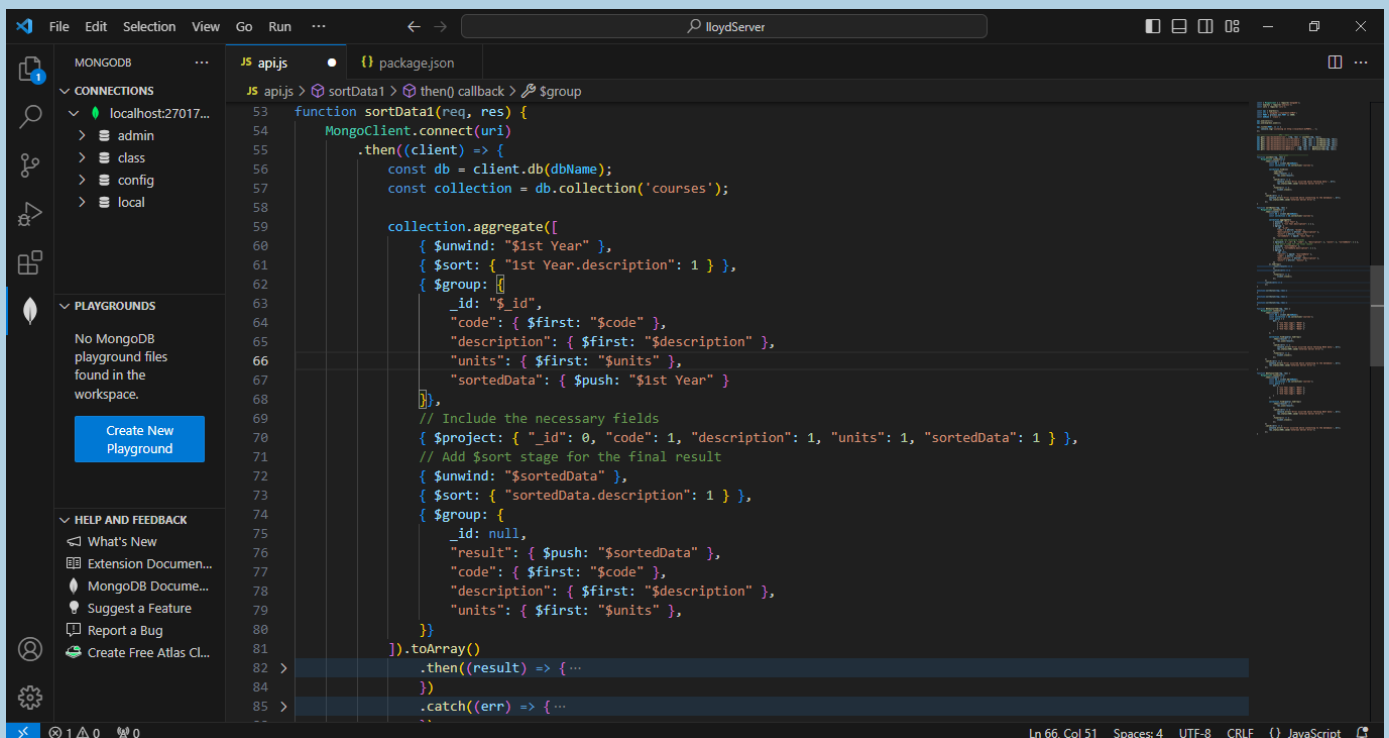




The screenshot shows the VS Code editor interface with the MongoDB extension. The left sidebar displays the 'CONNECTIONS' panel with a connection to 'localhost:27017...' and a 'PLAYGROUNDS' panel with a 'Create New Playground' button. The main editor area shows a JavaScript file named 'apijs' with the following code:

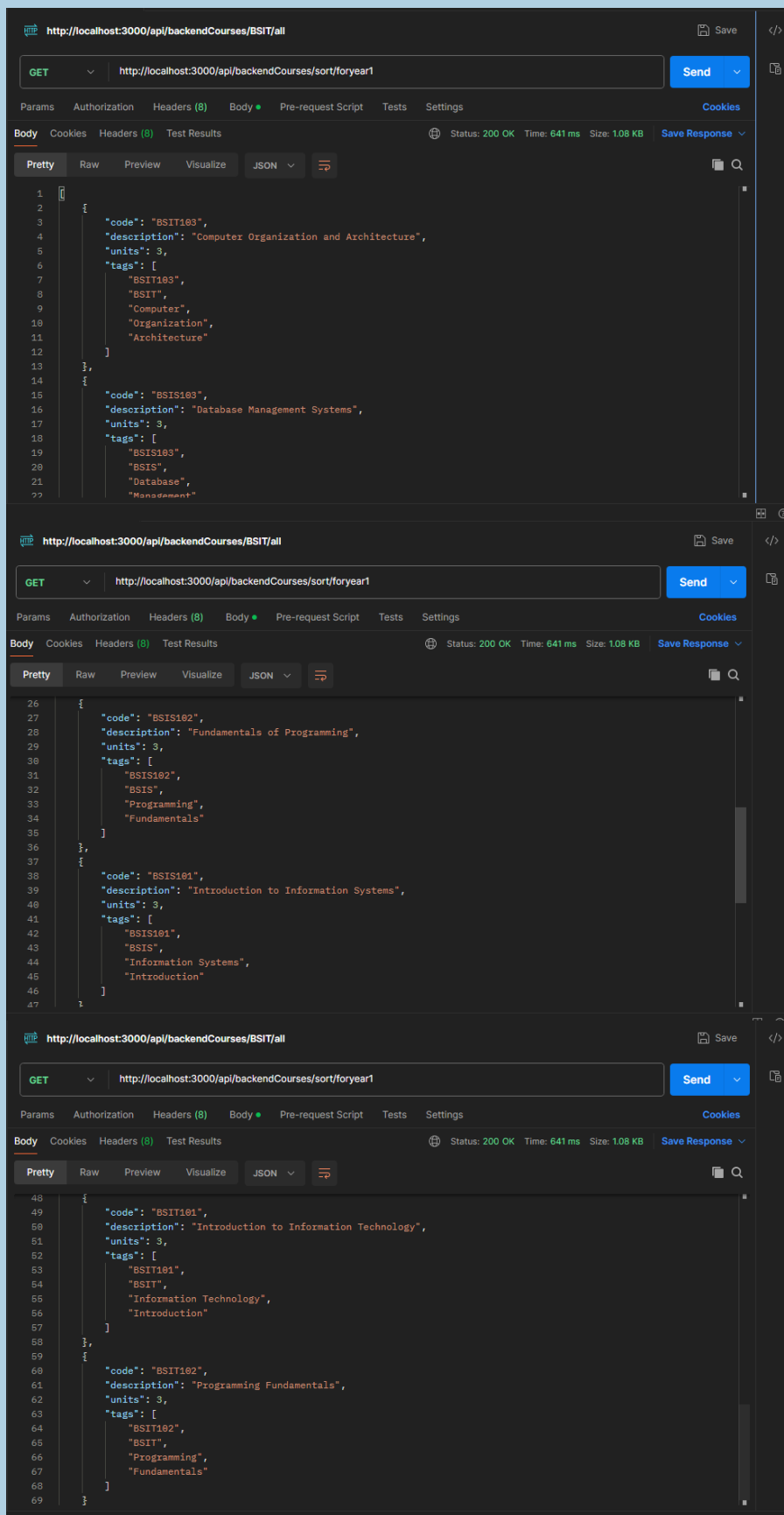
```
28 function justGet(req, res) {
29   .then((client) => {
30     //
43     .finally(() => {
44       client.close();
45     });
46   })
47   .catch((err) => {
48     console.error('Error occurred while connecting to the database:', err);
49     res.status(500).send('Internal Server Error');
50   });
51 }
52
53 > function sortData1(req, res) { ...
97 }
98
99 > function sortData2(req, res) { ...
143 }
144
145 > function sortData3(req, res) { ...
189 }
190
191 > function sortData4(req, res) { ...
235 }
236
237 function BSISunsorted(req, res) {
238   MongoClient.connect(uri)
239   .then((client) => {
240     const db = client.db(dbName);
241     const collection = db.collection('courses');
242     const query = {
243       $or: [
244         { "1st Year.tags": "BSIS" },
```

Here is the sorted data by year level



The screenshot shows the VS Code editor interface with the MongoDB extension. The left sidebar displays the 'CONNECTIONS' panel with a connection to 'localhost:27017...' and a 'PLAYGROUNDS' panel with a 'Create New Playground' button. The main editor area shows a JavaScript file named 'apijs' with the following code:

```
53 function sortData1(req, res) {
54   MongoClient.connect(uri)
55   .then((client) => {
56     const db = client.db(dbName);
57     const collection = db.collection('courses');
58
59     collection.aggregate([
60       { $unwind: "$1st Year" },
61       { $sort: { "1st Year.description": 1 } },
62       { $group: {
63         _id: "$_id",
64         "code": { $first: "$code" },
65         "description": { $first: "$description" },
66         "units": { $first: "$units" },
67         "sortedData": { $push: "$1st Year" }
68       } }],
69       // Include the necessary fields
70       { $project: { "_id": 0, "code": 1, "description": 1, "units": 1, "sortedData": 1 } },
71       // Add $sort stage for the final result
72       { $unwind: "$sortedData" },
73       { $sort: { "sortedData.description": 1 } },
74       { $group: {
75         _id: null,
76         "result": { $push: "$sortedData" },
77         "code": { $first: "$code" },
78         "description": { $first: "$description" },
79         "units": { $first: "$units" },
80       } }],
81       .toArray()
82     .then((result) => { ...
84     })
85     .catch((err) => { ...
```



Each year level has a sorter.

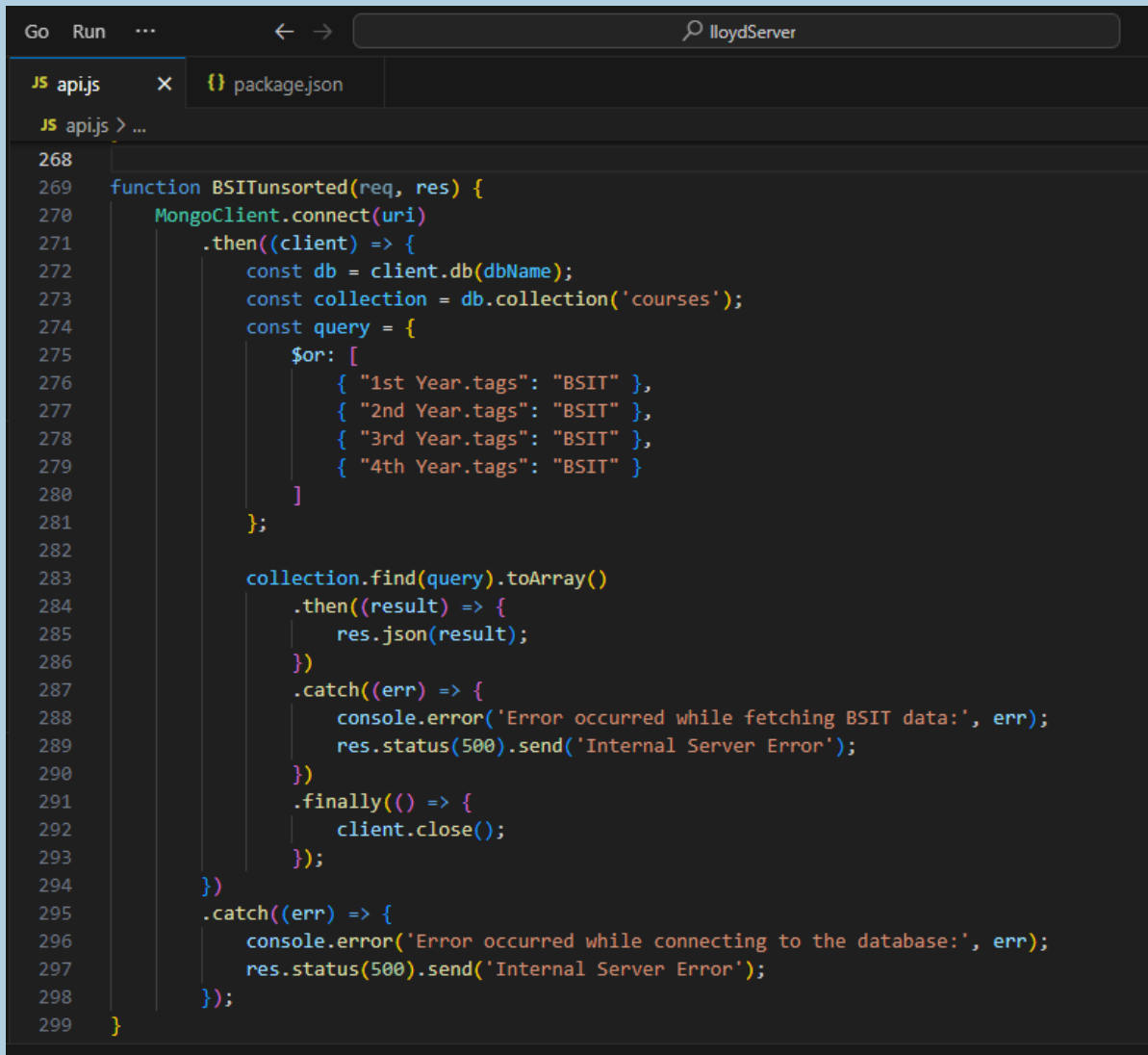
This is the Code for Retrieve all published BSIS courses from the curriculum.

```
Go Run ... ← → LloydServer
JS apijs x {} package.json
JS apijs > BSITunsorted
236
237 function BSISunsorted(req, res) {
238   MongoClient.connect(uri)
239     .then((client) => {
240       const db = client.db(dbName);
241       const collection = db.collection('courses');
242       const query = {
243         $or: [
244           { "1st Year.tags": "BSIS" },
245           { "2nd Year.tags": "BSIS" },
246           { "3rd Year.tags": "BSIS" },
247           { "4th Year.tags": "BSIS" }
248         ]
249       };
250
251       collection.find(query).toArray()
252         .then((result) => {
253           res.json(result);
254         })
255         .catch((err) => {
256           console.error('Error occurred while fetching BSIS data:', err);
257           res.status(500).send('Internal Server Error');
258         })
259         .finally(() => {
260           client.close();
261         });
262     })
263     .catch((err) => {
264       console.error('Error occurred while connecting to the database:', err);
265       res.status(500).send('Internal Server Error');
266     });
267 }
```

Here are the output:

```
http://localhost:3000/api/backendCourses/BSIT/all
GET http://localhost:3000/api/backendCourses/BSIT/all
Status: 200 OK Time: 115 ms Size: 153 KB Save Response
Pretty Raw Preview Visualize JSON
{
  "_id": "65d70bb6025fe9c027c355e7",
  "1st Year": [
    {
      "code": "BSIT101",
      "description": "Introduction to Information Technology",
      "units": 3,
      "tags": [
        "BSIT101",
        "BSIT",
        "Information Technology",
        "Introduction"
      ]
    },
    {
      "code": "BSIT102",
      "description": "Programming Fundamentals",
      "units": 3,
      "tags": [
        "BSIT102",
        "BSIT",
        "Programming"
      ]
    }
  ]
}
```

And thi is the Code for Retrieve all published BSIT courses from the curriculum.



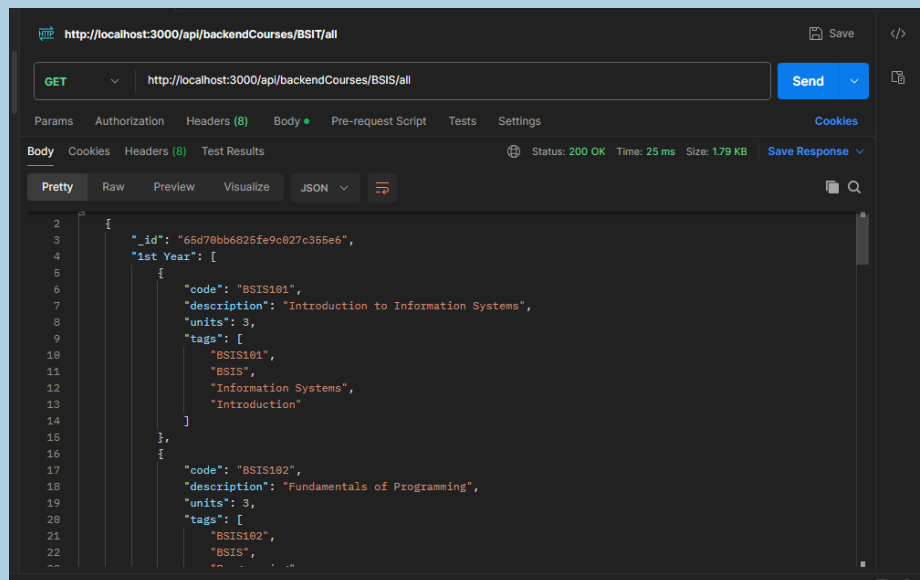
```
Go Run ...  ← →  lloydServer

JS api.js  X  {} package.json

JS api.js > ...

268
269 function BSITUnsorted(req, res) {
270   MongoClient.connect(uri)
271     .then((client) => {
272     const db = client.db(dbName);
273     const collection = db.collection('courses');
274     const query = {
275       $or: [
276         { "1st Year.tags": "BSIT" },
277         { "2nd Year.tags": "BSIT" },
278         { "3rd Year.tags": "BSIT" },
279         { "4th Year.tags": "BSIT" }
280       ]
281     };
282
283     collection.find(query).toArray()
284       .then((result) => {
285         res.json(result);
286       })
287       .catch((err) => {
288         console.error('Error occurred while fetching BSIT data:', err);
289         res.status(500).send('Internal Server Error');
290       })
291       .finally(() => {
292         client.close();
293       });
294   })
295   .catch((err) => {
296     console.error('Error occurred while connecting to the database:', err);
297     res.status(500).send('Internal Server Error');
298   });
299 }
```

Here are the output:



```
http://localhost:3000/api/backendCourses/BSIT/all

GET http://localhost:3000/api/backendCourses/BSIT/all Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

Body Cookies Headers (8) Test Results Status: 200 OK Time: 25 ms Size: 1.79 KB Save Response

Pretty Raw Preview Visualize JSON

2 {
3   "_id": "65d78bb6825fe9c027c355e6",
4   "1st Year": [
5     {
6       "code": "BSIS101",
7       "description": "Introduction to Information Systems",
8       "units": 3,
9       "tags": [
10        "BSIS101",
11        "BSIS",
12        "Information Systems",
13        "Introduction"
14      ]
15    },
16    {
17      "code": "BSIS102",
18      "description": "Fundamentals of Programming",
19      "units": 3,
20      "tags": [
21        "BSIS102",
22        "BSIS",
23        "Fundamentals of Programming",
24        "Introduction"
25      ]
26    }
27  ]
28 }
```